

## UIC Editing VBA Code

Contributed by Candace Cady  
25, Oct. 2007  
Last Updated 29, Nov. 2007

VBA Code to be used within an edit session to autopopulate the CreatedOn, ModifiedOn, EditedBy, and certain UIC program unique identifiers (well, facility, and authorization numbers). Code written by Bert Granberg.

### Option Explicit

```
Private m_bPopulate As Boolean
Private m_IsEditEventChange As Boolean
Private WithEvents m_pEditEvents As Editor
```

```
Private m_pMxDoc As IMxDocument
Private m_pMap As IMap
Private m_pSDEFWS As IFeatureWorkspace
Private m_pEditor As IEditor
Private Function MxDocument_OpenDocument() As Boolean
'initialize custom editing environment variables so that an onchangefeature and oncreatefeature event will run
Dim pEditor As IEditor
Dim pUID As New UID
Set m_pMxDoc = ThisDocument
Set m_pMap = m_pMxDoc.FocusMap
pUID = "esriCore.Editor"
Set m_pEditor = Application.FindExtensionByCLSID(pUID)
If m_pEditor Is Nothing Then
MsgBox "Unable to Enable UIC Custom Editing Environment", vbOKOnly, "ERROR...UIC Editing Environment"
Exit Function
End If
Set m_pEditEvents = m_pEditor
m_bPopulate = True
m_IsEditEventChange = False
End Function
```

```
Private Sub m_pEditEvents_OnChangeFeature(ByVal obj As esriGeoDatabase.IObject)
'custom behavior when uic features and table rows are modified. program id's are calculated from
'the right characters of the GUID primary key for authorization, facility, and well. for all features
'and table rows and modifiedon and editedby values are updated
```

```
If m_IsEditEventChange = False Then
Dim pRow As IRow
Dim createdOnFieldIndex As Integer
Dim modifiedOnFieldIndex As Integer
Dim editedByFieldIndex As Integer
Dim facilityIDFieldIndex As Integer
Dim authorizationIDFieldIndex As Integer
Dim wellIDFieldIndex As Integer
Dim facilityFKFieldIndex As Integer
Dim facilityFK As String
Dim guidFieldIndex As Integer
Dim guid_ID As String
Dim authorizationType As String
Dim authorizationTypeFieldIndex As Integer
Dim countyChars As String
Dim wellClass As Integer
Dim wellClassStr As String
Dim changeMade As Boolean
Dim wellClassFieldIndex As Integer
Dim pRowsDataset As IDataset
```

```
Set pRow = obj
```

```
If Not m_bPopulate Then Exit Sub
```

```
createdOnFieldIndex = pRow.Fields.FindField("CreatedOn")
modifiedOnFieldIndex = pRow.Fields.FindField("ModifiedOn")
editedByFieldIndex = pRow.Fields.FindField("EditedBy")
facilityIDFieldIndex = pRow.Fields.FindField("FacilityID")
authorizationIDFieldIndex = pRow.Fields.FindField("AuthorizationID")
wellIDFieldIndex = pRow.Fields.FindField("WellID")
guidFieldIndex = pRow.Fields.FindField("GUID")
facilityFKFieldIndex = pRow.Fields.FindField("Facility_FK")
wellClassFieldIndex = pRow.Fields.FindField("WellClass")
```

```
changeMade = False
```

```
If guidFieldIndex > 0 Then
```

```
  If Not IsNull(pRow.Value(guidFieldIndex)) Then
```

```
    guid_ID = pRow.Value(guidFieldIndex)
```

```
    If authorizationIDFieldIndex > 0 Then
```

```
      authorizationTypeFieldIndex = pRow.Fields.FindField("AuthorizationType")
```

```
      If authorizationTypeFieldIndex > 0 Then
```

```
        If Not IsNull(pRow.Value(authorizationTypeFieldIndex)) Then
```

```
          authorizationType = Left(pRow.Value(authorizationTypeFieldIndex), 2)
```

```
          If Len(authorizationType) <> 2 Then
```

```
            authorizationType = "XX" 'no code available
```

```
            MsgBox "Warning, the AuthorizationType set for this authorization record is not the correct length (should be 2 characters). The AuthorizationID will carry an XX value until the AuthorizationType is set.", vbOKOnly
```

```
          End If
```

```
        Else
```

```
          authorizationType = "XX" 'no code available
```

```
          MsgBox "Warning, no AuthorizationType set for this authorization record. The AuthorizationID will carry an XX value until the AuthorizationType is set.", vbOKOnly
```

```
        End If
```

```
        countyChars = "CC" 'CC is default value in case county fips is unknown
```

```
      If facilityFKFieldIndex > 0 Then
```

```
        If Not IsNull(pRow.Value(facilityFKFieldIndex)) Then
```

```
          facilityFK = pRow.Value(facilityFKFieldIndex)
```

```
          countyChars = GetCountyFIPSFFromFacilityFC(pRow.Table, facilityFK)
```

```
        End If
```

```
      End If
```

```
      pRow.Value(authorizationIDFieldIndex) = "UTU" & countyChars & authorizationType & Left(Right(guid_ID, 8), 7)
```

```
      changeMade = True
```

```
    End If
```

```
  End If
```

```
If wellIDFieldIndex > 0 Then
```

```
  wellClassStr = "0"
```

```
  If wellClassFieldIndex > 0 Then
```

```
    If Not IsNull(pRow.Value(wellClassFieldIndex)) Then
```

```
      wellClass = pRow.Value(wellClassFieldIndex)
```

```
      If wellClass < 1 And wellClass > 10 Then
```

```
        wellClassStr = "0" 'no code available
```

```
      Else
```

```
        wellClassStr = CStr(wellClass)
```

```
      End If
```

```
    End If
```

```
    countyChars = "CC" 'CC is default value in case county fips is unknown
```

```
  If facilityFKFieldIndex > 0 Then
```

```
    If Not IsNull(pRow.Value(facilityFKFieldIndex)) Then
```

```
      facilityFK = pRow.Value(facilityFKFieldIndex)
```

```

        countyChars = GetCountyFIPSFromFacilityFC(pRow.Table, facilityFK)
    End If
End If

pRow.Value(wellIDFieldIndex) = "UTU" & countyChars & wellClass & Left(Right(guid_ID, 9), 8)
changeMade = True
End If
End If

If facilityIDFieldIndex > 0 Then

    countyChars = "CC" 'CC is default value in case county fips is unknow
    If pRow.Fields.FindField("CountyFIPS") > 0 Then
        If Not IsNull(pRow.Value(pRow.Fields.FindField("CountyFIPS"))) Then
            countyChars = Right(pRow.Value(pRow.Fields.FindField("CountyFIPS")), 2)
        End If
    End If

    Dim countyFIPSFieldIndex As Integer
    Dim countyFIPSStr As String
    Dim pFacDataset As IDataset
    Dim pFeatureWorkspace As IFeatureWorkspace
    Dim pWellTable As ITable
    Dim pAuthorizationTable As ITable
    'Set pFacDataset = pRow.Table
    'Set pFeatureWorkspace = pFacDataset.Workspace
    'Dim pVersion As IVersion
    'Set pVersion = pFeatureWorkspace
    'Set pWellTable = pFeatureWorkspace.OpenTable("UICWell")

    Dim pTC As ITableCollection
    Set pTC = m_pMap

    Dim pEnumLayer As IEnumLayer
    Dim pLayer As ILayer
    Dim pFeatureLayer As IFeatureLayer
    Dim pFLFeatureClass As IFeatureClass
    Dim pFLDataset As IDataset

    Set pEnumLayer = m_pMap.Layers
    Set pLayer = pEnumLayer.Next

    Do Until pLayer Is Nothing
        If TypeOf pLayer Is IFeatureLayer Then
            Set pFeatureLayer = pLayer
            Set pFLFeatureClass = pFeatureLayer.FeatureClass
            Set pFLDataset = pFLFeatureClass 'QI

            Debug.Print pFLDataset.Name

            Set pRowsDataset = pRow.Table 'QI
            If InStr(pFLDataset.Name, "UICWell") > 0 And m_pEditor.EditWorkspace.PathName =
pRowsDataset.Workspace.PathName Then
                Set pWellTable = pFLFeatureClass
            Exit Do
        End If
    End If
    Set pLayer = pEnumLayer.Next
Loop

    Dim t As Integer
    Dim pTable As ITable
    Dim dt As IDataset

```

```

For t = 0 To pTC.TableCount - 1
  If Not pTC.Table(t) Is Nothing Then
    Set pTable = pTC.Table(t)
    Set dt = pTable
    Debug.Print dt.Name
    Set pRowsDataset = pRow.Table 'QI
    If InStr(pFLDataset.Name, "UICAuthorization") > 0 And m_pEditor.EditWorkspace.PathName =
pRowsDataset.Workspace.PathName Then

      'If dt.Name = "UDEQ.UICADMIN.UICAuthorization" Then
        Set pAuthorizationTable = pTable
        Exit For
      End If
    End If
  End If
Next t

countyFIPSFieldIndex = pRow.Fields.FindField("CountyFIPS")

If countyFIPSFieldIndex > 0 Then
  countyFIPSStr = "CC"
  If Not IsNull(pRow.Value(countyFIPSFieldIndex)) Then
    countyFIPSStr = Right(pRow.Value(countyFIPSFieldIndex), 2)
  End If
  If Not pAuthorizationTable Is Nothing Then
    Call changeAuthorizationIDinTable(guid_ID, countyFIPSStr, pAuthorizationTable)
  End If
  If Not pWellTable Is Nothing Then
    Call changeWellIDinTable(guid_ID, countyFIPSStr, pWellTable)
  End If
End If
pRow.Value(facilityIDFieldIndex) = "UTU" & countyChars & "F" & Left(Right(guid_ID, 9), 8)
changeMade = True
End If
Else
  MsgBox "Warning, a NULL value missing in the required GUID field. Unable to update related records.", vbOKOnly
End If
End If

If modifiedOnFieldIndex > 0 Then
  pRow.Value(modifiedOnFieldIndex) = Now
  changeMade = True
End If

If editedByFieldIndex > 0 Then
  pRow.Value(editedByFieldIndex) = Environ("USERNAME")
  changeMade = True
End If

If changeMade Then
  m_IsEditEventChange = True
  pRow.Store
  m_IsEditEventChange = False
End If
End If

End Sub

Private Sub m_pEditEvents_OnCreateFeature(ByVal obj As esriGeoDatabase.IObject)
'on create feature code runs for new UIC features and table rows to populate the primary key
'with a GUID value and to set createon, modifiedon, and editedby attributes.

Dim pRow As IRow
Dim createdOnFieldIndex As Integer
Dim modifiedOnFieldIndex As Integer
Dim editedByFieldIndex As Integer

```

```
Dim pDataset As IDataset
'Dim facilityFKFieldIndex As Integer
'Dim facilityFK As String
Dim guidFieldIndex As Integer
Dim newGUID As IUID

'Dim authorizationType As String
'Dim authorizationTypeFieldIndex As Integer
'Dim authorizationIDFieldIndex As Integer
'Dim countyChars As String

Dim changeMade As Boolean

Set pRow = obj

If Not m_bPopulate Then Exit Sub

createdOnFieldIndex = pRow.Fields.FindField("CreatedOn")
modifiedOnFieldIndex = pRow.Fields.FindField("ModifiedOn")
editedByFieldIndex = pRow.Fields.FindField("EditedBy")
'authorizationIDFieldIndex = pRow.Fields.FindField("AuthorizationID")
guidFieldIndex = pRow.Fields.FindField("GUID")
'facilityFKFieldIndex = pRow.Fields.FindField("Facility_FK")

changeMade = False

If guidFieldIndex > 0 Then
    Set newGUID = New UID
    newGUID.Generate
    pRow.Value(guidFieldIndex) = newGUID.Value
    changeMade = True
End If

If modifiedOnFieldIndex > 0 Then
    pRow.Value(modifiedOnFieldIndex) = Now
    changeMade = True
End If

If createdOnFieldIndex > 0 Then
    pRow.Value(createdOnFieldIndex) = Now
    changeMade = True
End If

If editedByFieldIndex > 0 Then
    pRow.Value(editedByFieldIndex) = Environ("USERNAME")
    changeMade = True
End If

If changeMade Then
    m_IsEditEventChange = True
    pRow.Store
    m_IsEditEventChange = False
End If

Set pDataset = pRow.Table
If Right(pDataset.Name, 11) = "UICFacility" Then
    MsgBox "Use the ArcMap Attribute Editor to choose a value for the CountyFIPS field." _
        & vbNewLine & "Populating the CountyFIPS code will result in this new feature being" _
        & vbNewLine & "assigned a FacilityID.", vbOKOnly, "UIC Reminder...Facilities"
ElseIf Right(pDataset.Name, 16) = "UICAuthorization" Then
    MsgBox "Use the ArcMap Attribute Editor to change the AuthorizationType field value." _
        & vbNewLine & "Changing the AuthorizationType value will result in this new record being" _
        & vbNewLine & "assigned a AuthorizationID.", vbOKOnly, "UIC Reminder...Authorizations"
ElseIf Right(pDataset.Name, 7) = "UICWell" Then
    MsgBox "Use the ArcMap Attribute Editor to choose a value for the WellClass field." _
```

```

    & vbNewLine & "Populating the WellClass field will result in this new feature being" _
    & vbNewLine & "assigned a WellID.", vbOKOnly, "UIC Reminder...Wells"
End If

```

```
End Sub
```

```
Public Function GetCountyFIPSFromFacilityFC(pInTable As ITable, queryFKStr As String) As String
```

```
'get the county fips from the facility feature class, given a facility foreign key
```

```
GetCountyFIPSFromFacilityFC = "CC"
```

```
Dim pInDataset As IDataset
```

```
Dim pFeatureWorkspace As IFeatureWorkspace
```

```
Dim pFacilityTable As ITable
```

```
Dim pQF As IQueryFilter
```

```
Dim pCursor As ICursor
```

```
Dim pRow As IRow
```

```
Dim countyFIPSFieldIndex As Integer
```

```
Set pInDataset = pInTable
```

```
'Set pFeatureWorkspace = pInDataset.Workspace
```

```
'Set pFacilityTable = pFeatureWorkspace.OpenTable("UICFacility")
```

```
Dim pEnumLayer As IEnumLayer
```

```
Dim pLayer As ILayer
```

```
Dim pFeatureLayer As IFeatureLayer
```

```
Dim pFLFeatureClass As IFeatureClass
```

```
Dim pFLDataset As IDataset
```

```
Set pEnumLayer = m_pMap.Layers
```

```
Set pLayer = pEnumLayer.Next
```

```
Do Until pLayer Is Nothing
```

```
    If TypeOf pLayer Is IFeatureLayer Then
```

```
        Set pFeatureLayer = pLayer
```

```
        Set pFLFeatureClass = pFeatureLayer.FeatureClass
```

```
        Set pFLDataset = pFLFeatureClass 'QI
```

```
        Debug.Print pFLDataset.Name
```

```
        If InStr(pFLDataset.Name, "UICFacility") > 0 And m_pEditor.EditWorkspace.PathName =
pInDataset.Workspace.PathName Then
```

```
            'If pFLDataset.Name = "UDEQ.UICADMIN.UICFacility" Then
```

```
                Set pFacilityTable = pFLFeatureClass
```

```
                Exit Do
```

```
            End If
```

```
        End If
```

```
        Set pLayer = pEnumLayer.Next
```

```
Loop
```

```
countyFIPSFieldIndex = pFacilityTable.FindField("CountyFIPS")
```

```
If countyFIPSFieldIndex > 0 Then
```

```
    Set pQF = New QueryFilter
```

```
    pQF.WhereClause = "GUID = " & queryFKStr & ""
```

```
    Set pCursor = pFacilityTable.Search(pQF, False)
```

```
    Set pRow = pCursor.NextRow
```

```
    If Not pRow Is Nothing Then
```

```
        If Not IsNull(pRow.Value(countyFIPSFieldIndex)) Then
```

```
            GetCountyFIPSFromFacilityFC = Right(pRow.Value(countyFIPSFieldIndex), 2)
```

```
        End If
```

```

    End If
  End If
End Function

```

```

Public Sub changeAuthorizationIDinTable(inFacilityGUID As String, inCountyFIPS As String, inTable As ITable)
  'if a county fips attribute changes in the facility feature class, update the authorizationid which includes a
  'county fips component

```

```

  Dim pQF As IQueryFilter
  Dim pCursor As ICursor
  Dim pRow As IRow
  Dim authorizationIDFieldIndex As Integer
  Dim authorizationTypeFieldIndex As Integer
  Dim guidFieldIndex As Integer
  Dim facilityFKFieldIndex As Integer
  Dim prevAuth As String
  Dim currAuth As String

```

```

  authorizationIDFieldIndex = inTable.FindField("AuthorizationID")
  authorizationTypeFieldIndex = inTable.FindField("AuthorizationType")
  guidFieldIndex = inTable.FindField("GUID")
  facilityFKFieldIndex = inTable.FindField("Facility_FK")

```

```

  Dim guid_ID As String
  Dim authorizationType As String

```

```

  If authorizationTypeFieldIndex >= 0 And authorizationIDFieldIndex >= 0 And _
  facilityFKFieldIndex >= 0 And guidFieldIndex >= 0 Then

```

```

    Set pQF = New QueryFilter
    pQF.WhereClause = "Facility_FK = " & inFacilityGUID & ""

```

```

    Set pCursor = inTable.Search(pQF, False)
    Set pRow = pCursor.NextRow
    Do Until pRow Is Nothing

```

```

      prevAuth = ""
      If Not IsNull(pRow.Value(authorizationIDFieldIndex)) Then
        prevAuth = pRow.Value(authorizationIDFieldIndex)
      End If

```

```

      guid_ID = pRow.Value(guidFieldIndex)

```

```

      authorizationType = "XX"
      If Not IsNull(pRow.Value(authorizationTypeFieldIndex)) Then
        authorizationType = Left(pRow.Value(authorizationTypeFieldIndex), 2)
        If Len(authorizationType) <> 2 Then
          authorizationType = "XX" 'no code availab
        End If
      End If

```

```

      currAuth = "UTU" & inCountyFIPS & authorizationType & Left(Right(guid_ID, 8), 7)
      If prevAuth <> currAuth Then
        pRow.Value(authorizationIDFieldIndex) = currAuth
        m_IsEditEventChange = True
        pRow.Store
        m_IsEditEventChange = False
      End If

```

```

      Set pRow = pCursor.NextRow
    Loop
  End If

```

```

End Sub

```

```

Public Sub changeWellIDinTable(inFacilityGUID As String, inCountyFIPS As String, inTable As ITable)
    'if a county fips attribute changes in the facility feature class, update the well id which includes a
    'county fips component

    Dim pQF As IQueryFilter
    Dim pCursor As ICursor
    Dim pRow As IRow
    Dim wellIDFieldIndex As Integer
    Dim wellClassFieldIndex As Integer
    Dim guidFieldIndex As Integer
    Dim facilityFKFieldIndex As Integer
    Dim prevWell As String
    Dim currWell As String

    wellIDFieldIndex = inTable.FindField("WellID")
    wellClassFieldIndex = inTable.FindField("WellClass")
    guidFieldIndex = inTable.FindField("GUID")
    facilityFKFieldIndex = inTable.FindField("Facility_FK")

    Dim guid_ID As String
    Dim wellClass As String

    If wellClassFieldIndex >= 0 And wellIDFieldIndex >= 0 And _
        facilityFKFieldIndex >= 0 And guidFieldIndex >= 0 Then

        Set pQF = New QueryFilter
        pQF.WhereClause = "Facility_FK = " & inFacilityGUID & ""

        Set pCursor = inTable.Search(pQF, False)
        Set pRow = pCursor.NextRow

        Do Until pRow Is Nothing

            prevWell = ""
            If Not IsNull(pRow.Value(wellIDFieldIndex)) Then
                prevWell = pRow.Value(wellIDFieldIndex)
            End If

            guid_ID = pRow.Value(guidFieldIndex)

            wellClass = "X"
            If Not IsNull(pRow.Value(wellClassFieldIndex)) Then
                wellClass = Left(pRow.Value(wellClassFieldIndex), 1)
                If Len(wellClass) <> 1 Then
                    wellClass = "X" 'no code availab
                End If
            End If

            currWell = "UTU" & inCountyFIPS & wellClass & Left(Right(guid_ID, 9), 8)
            If prevWell <> currWell Then
                pRow.Value(wellIDFieldIndex) = currWell
                m_IsEditEventChange = True
                pRow.Store
                m_IsEditEventChange = False
            End If
            Set pRow = pCursor.NextRow
        Loop
    End If

End Sub

```